ACQUISITION OF BUILDING GEOMETRY IN THE SIMULATION OF ENERGY PERFORMANCE

Vladimir Bazjanac
Lawrence Berkeley National Laboratory
Berkeley, CA 94720 – U.S.A.

ABSTRACT

Building geometry is essential to any simulation of building performance. This paper examines from the users' point of view the importing of building geometry into simulation of energy performance. It discusses the basic options in moving from two- to three-dimensional definition of geometry and the ways to import that geometry into energy simulation. The obvious answer lies in software interoperability. With the BLIS group of interoperable software one can interactively import building geometry from CAD into EnergyPlus and dramatically reduce the effort needed for manual input.

INTRODUCTION

The current standard practice in preparing energy simulation input typically involves repetitive manual operation that in essence amounts to duplication of already existing data. The process is error-prone and the resulting simulation input code is difficult to debug. As the complexity of the building and the simulation increase, input preparation becomes more and more the main catalyst for abandoning (or not even starting) the simulation project.

The largest portion of the effort to prepare simulation input is absorbed by the definition of building geometry. Because few buildings are drawn or defined in 3-D, the complete set of information needed to define the building is usually distributed over a large number of 2-D drawings; this requires a substantial effort to comprehend and extract all the pertinent information.

Most architects and engineers depend on the use of some "mission-critical" software in their work. In the course of design of a building, building geometry may get recreated as much as seven or eight times: Structural, mechanical, and electrical engineering, as well as plumbing, energy calculation, lighting, code checking and cost estimating software all depend on building geometry information to do their work. In most cases building geometry is completely regenerated because one cannot import the needed definitions directly from CAD files that contain the original information.

When budgeting for building energy performance simulation, one can use the rule of thumb that says that the cost of input preparation and the cost of analysis of results should be approximately the same; relative to these, the cost of simulation runs (i.e., computer run management and computer time) is minimal (Figure 1).

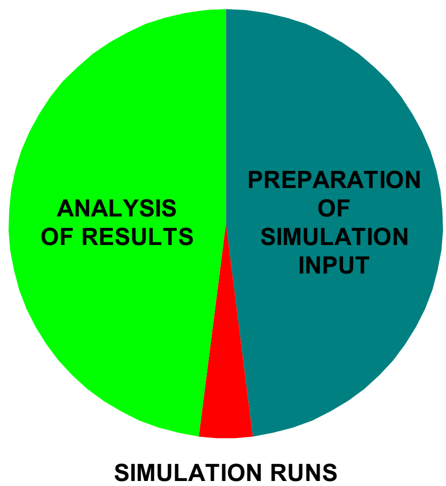

Figure 1 – Distribution of effort in a building energy performance simulation project

Most of the effort in the preparation of simulation input is in getting the first successful run (Figure 2). The process that consists of input definition, debugging, and computer runs and analysis of results is repetitive and based on feedback; it often takes many iterations before the result is satisfactory. Subsequent additions and modifications to simulation input that may be needed for parametric runs require comparatively little effort.

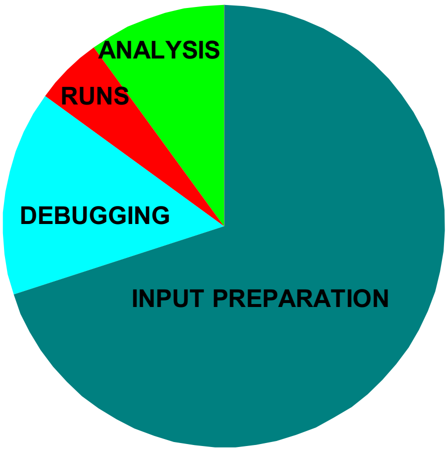

Figure 2 – Distribution of effort that yields a successful simulation run

In the case of building energy performance simulation, up to 80% of the effort in input preparation may be consumed on the definition of building geometry (Figure 3). By definition, most of the building geometry must be defined for the first successful run. The *actual* distribution of effort *varies greatly* from building to building for several reasons: It depends on the complexity and size of the building and its geometry, on the purpose and goals of the simulation, on the expertise and experience of those who are designing the simulation and preparing the input, on the computer aids that are used in the process, on the schedule and budget, and on several other factors that may affect the case.

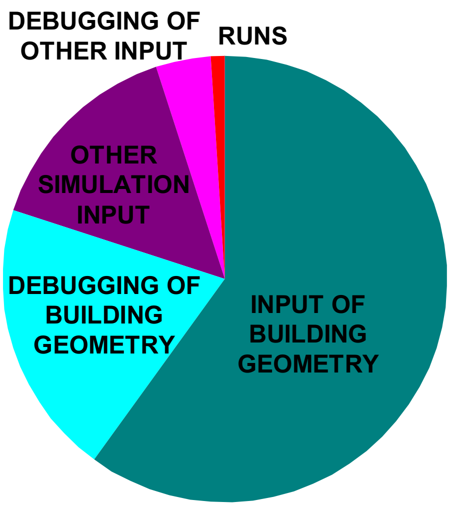

Figure 3 –Building geometry as portion of overall effort to prepare simulation input

Manual input of building geometry and debugging require continuous high level of concentration and consistency. It is a tedious process that can result in frustration. It tempts one to resort to "approximation of convenience" to "get something running" sooner; that can cause very serious difficulties later and possibly compromise the entire simulation effort.

In reality, the simulation of building energy/thermal performance is often not used the way simulation is supposed to work in its classical sense: to perform multiple experiments and rely on statistical analysis of results to determine meaningful future outcomes [Naylor et al. 1966]. All too often the investigation of alternatives is limited to one or only a few simulation runs and the results are accepted as definitive answers. Many factors are responsible for that; one can argue that the effort and cost of acquisition of building geometry and the associated high cost of simulation input preparation play a prominent role.

Those who prepare input for simulation models and use them often dream of tools that could import building geometry automatically. While the completely automatic acquisition may never be achieved, it is now possible to *partially* automate the process and significantly reduce the effort and its cost.

This paper discusses the acquisition of building geometry from the point of view of the *user* of building energy performance simulation. It defines issues from that point of view and discusses some of the options the user has today, with the hope that the discussion will lead to less frustration in importing complex geometry into simulation projects. The paper does not even attempt to deal with related theoretical and technical (software) issues that have been addressed elsewhere.

MOVING FROM 2-D TO 3-D

In regard to the representation of building geometry, sophisticated whole-building energy performance simulation tools are essentially three-dimensional. They all employ three-dimensional coordinate systems and require spatial definitions. BLAST, DOE-2, EnergyPlus and ESP-r are no exception.

While more and more architects are defining and documenting their building designs as three-dimensional models, most buildings are still documented in form of two-dimensional drawings. There is even sporadic evidence in Europe that some of the early "converts" to 3-D are reverting back to line drawing [Haas 2001].

To an individual that is preparing the building geometry part of input for energy performance simulation, this poses a problem of converting building geometry contained in line drawings to an at least semi-intelligent building model representation. While others have amply discussed this problem, that individual actually has few choices how to proceed:

1. Interpret the drawings, scale off dimensions and manually key in values that define the location and size of building elements in question, all according to the rules and syntax of the particular simulation tool. This is the most frequently used method to-date. It does not take advantage of software interoperability, and is time consuming and very prone to error.

2. Use a simulation tool with a Graphic User Interface (GUI) that facilitates the definition of the specific building's geometry for the simulation. This presumes that (a) such a tool is readily available and (b) that the particular GUI is capable of adequately dealing with all the complexities of the particular building's geometry. If the tool's geometry definition via the GUI is based on stencils, it is likely that the resulting geometry definitions will be adequate only for the cases where the building layout matches the stencil's layout reasonably well. The benefit of this method is that, once the building geometry is laid out, the user no longer has to manually neither enter the data nor deal with the tool's related syntax.

3. Convert the representation of building geometry from two- to three-dimensional *before* it is reformulated for input for the simulation. The difficulties here are that (a) the market currently offers no software that can perform the task effortlessly, (b) the conversion to 3-D typically contains much more information than needed for the simulation, and (c) one is still left with the task of importing the now three-dimensional geometry into the simulation.

A number of CAD tools can almost automatically generate some sort of a 3-D building representation from 2-D drawings. The approach of some of these tools is rather ingenious: They "prop up" building elevations at the perimeter of the floor plan to create a three-dimensional representation of the building envelope, or they create three-dimensional models by drawing edges of volumes defined in two-dimensional drawings. Unfortunately, such 3-D models of building geometry are not objectified from a building data point of view and are not very useful in thermal simulation.

Potentially much more useful are tools that generate 3-D representations of building geometry from 2-D drawings *interactively*. Such tools require a substantial amount of partially-automated-partially-manual recreation of the third dimension on top of 2-D representation, but they also typically provide an opportunity to objectify the definitions. A number of such tools has lately emerged on the market.

It is important to remember that manual extraction of information from two-dimensional drawings is always based on *human interpretation*. As much as most of line drawing follows well established conventions of what to represent how, the full three-dimensional understanding of space and objects results from combining information contained in several sources (e.g., building floor plans, sections and elevations). If the consideration of multiple sources is not thorough and tedious, mistakes and misrepresentations are possible and sometimes even likely. That makes the extraction of the third dimension from 2-D drawings sometimes a difficult process; the degree of difficulty clearly increases with the complexity of building geometry.

IMPORT OF GEOMETRY DATA

Naturally, if the building geometry is originally defined in 3-D, all afore mentioned issues vanish except for one: how to import the geometry into the simulation. To do that "seamlessly" (i.e., import building geometry directly from its source without human intervention) one needs an interface between the simulation tool and the source, generator or the container of geometry. In each instance that interface must be able to understand the data structure of both the source application or the database *and* the simulation tool, and must be capable of translating the source information according to the rules and syntax of the simulation tool. Given that this is no small feat and that such interfaces are typically "dedicated" (i.e., they interface only two or a very small number of specific software applications or data bases), this is an expensive and relatively rare solution.

An alternative is to use a simulation tool with a GUI or a pre-processor that can accept 3-D definitions and transform them into parts of simulation input. Unfortunately, GUIs and pre-processors designed for simulation of building energy performance and currently on the market either cannot directly import complete 3-D definitions of buildings or require manual editing of simulation input.

As always, one can use the manual solution: reading the information from the 3-D model and keying it in according to the rules and syntax of the simulation tool. But the probability of error increases with the volume of manual input preparation.

A "simulation view" of the building is different from the architects' and engineers'. The thermal simulation view of building geometry contains much less information but may demand peculiar detail that may not have been defined in the architects' view. Some walls, windows and doors, as well as some rooms or spaces may be completely omitted, for example. Yet other walls may have to be subdivided. All such differences must be reflected in the definition of building geometry that is imported into simulation. The following section discusses some of issues that are quite typical for the definition of building geometry for simulation of thermal performance.

REASONS FOR INTERVENTION

A fully developed architectural definition of building geometry contains a lot more information than is needed for building energy performance simulation. The reduction of that information mostly involves simplification that requires human judgment and intervention.

The most common case is elimination of those parts of building geometry that are irrelevant to the simulation. Interior walls, windows and doors between spaces in the same thermal zone that maintain the same temperature conduct no thermal transfer; these and other building parts that have no effect on the simulation can be omitted.

Sometimes many repetitive elements (e.g., multiple individual exterior shading surfaces) have to be grouped to expedite simulation execution. For the same reason, repetitive descriptions of identical spaces and surfaces (walls, windows, doors, etc.) are often defined once and then "multiplied."

More difficult are approximations that are needed because the simulation tool cannot deal with irregular shapes. The best example of that are curved walls and roofs. Few building energy performance simulation tools can define curved surfaces; when such are encountered, they have to be approximated with flat segments. The rules of segmentation vary from one case to another.

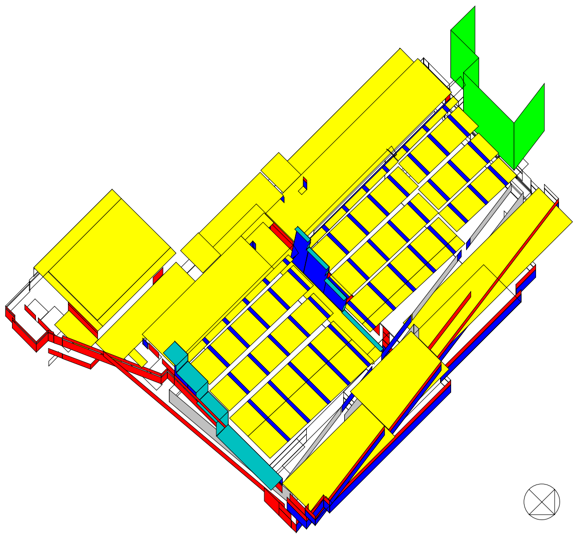

Figure 4 –Flat segments that approximate the curved roof and skylights of the new Pittsburgh Convention Center, as shown with DrawBDL

For example, Raphael Vignoly Architects PC designed the new David L. Lawrence Convention Center in Pittsburgh, PA with a roof that consisted of three very large curved surfaces. In the modeling of the building for simulation with DOE-2 the roof geometry had to be approximated with 70 flat horizontal segments (Figure 4). The approximation had to accommodate 14 long skylights positioned across the roof. In addition, the geometry of all glazing that extended to the roof had to be reconfigured to properly simulate the shading effect from the roof. Finding an acceptable approximation took more effort than the definition of the rest of that building's geometry.

Some of the information that is part of building geometry is sometimes not included in CAD drawings or must be developed specifically for the simulation. This could be (the 3-D) location of sensors perhaps endogenous to the simulation that monitor and control events in the simulation, such as (day)light sensors that control the use of electrical lighting. Or it could be the different coloring and surface treatment of otherwise the same wall that requires the subdivision of the wall for proper definition of reflectance at different locations. The missing information must be developed and added to the building geometry input.

Thermal zoning for simulation usually requires agglomeration of spaces that share the same thermal conditions and are operated in the same way. In other words, two or more spaces in the building are merged into one thermal zone. Given a complete HVAC design for the building, one would expect that this task could be automated and made an integral part of building geometry input for simulation. This is not the case, because decisions about thermal zoning may require the consideration of other factors, such as occupancy characteristics that may result in different internal schedules and loads in otherwise identical spaces. The consideration of daylighting also affects zoning decisions. Such cases require human judgment, decisions, modifications and additions to the original definitions of building geometry.

CAD tools have shortcomings too, regardless of how sophisticated they are. A few are not able to properly define all shapes and volumes one may encounter in a building. Others have apparently useful advanced features that are not adequately documented. And yet others have advanced features that have not been fully debugged. To effectively help in the preparation of input of geometry for simulation *all* require sophisticated, experienced users. Inexperienced users are better off not using these tools until they acquire sufficient skills.

GRAPHIC USER INTERFACE

While the number of available whole-building energy performance simulation engines is still quite small, the number of simulation tools on the market seems to be proliferating. Most new tools incorporate an existing simulation engine with a GUI and a post-processor that make the use of simulation in some way more convenient. Unfortunately, few of these GUIs are designed in a way that completely facilitates the definition or import of building geometry.

A GUI *truly* useful in the acquisition of building geometry should be able to:

- Deal with *any arbitrary* geometry, and deal with it *in 3-D*. This is a "non negotiable" requirement: While stencils are helpful and may save time when used, most complex buildings' geometry cannot be properly "shoed in."
- Read CAD files in their native format. That eliminates problems that arise from image translation, such as layer control, line color and weight, font type and point, etc., and makes working with the original information easier.
- Facilitate the use of layers and overlays. It is good practice to make all simplifications and additions to building geometry on separate layers and/or overlays. This pays off increasingly as the simulation input develops and expands.
- Support simultaneous display of multiple drawings and views. This saves time in detecting errors.
- Include a fully functional ASCII text editor that can expose the simulation tool's input syntax. That permits simple corrections or additions to the input that are often accomplished quicker with a text editor. It also facilitates the pasting of segments of input from other simulated buildings that is appropriate to reuse.
- Support cut-and-paste among multiple documents. This is useful when incorporating information from files in different format.
- Provide seamless access to external databases and libraries. Some of the building components

that have geometry are defined in manufacturers' and/or other databases and libraries and carry information in addition to geometry that can be used in the simulation.

In the absence of a useful GUI one can use virtually any sophisticated and fully functional CAD tool as a substitute. Current releases of such tools can perform all functions listed above as requirements for GUIs that assist in the acquisition of building geometry. The drawback is that the user must be highly skilled in the use of the particular CAD tool to use it for this purpose.

SOFTWARE INTEROPERABILITY

The obvious answer to acquisition of building geometry from CAD is *software interoperability*: direct exchange of data among different software applications. Such exchange requires a common data model that is shared (or at least "understood") by the exchanging applications [Bazjanac and Crawley 1999].

The idea is not new. The International Alliance for Interoperability (IAI) has been developing an objectified data model of buildings for more than six years [International Alliance for Interoperability 1999]. The data model is called the International Foundation Classes (IFC) and its latest version (IFC 2x) was released in October 2000. This data model fully supports the three-dimensional definition of building geometry. In addition to other information, software applications and tools that have implemented the IFC data model and are "downstream" in the design/analysis process can directly import building geometry that was generated by "upstream" CAD tools. They can also modify it and send it back "upstream."

A group of industry partners formed a joint project, the Building Lifecycle Interoperable Software (BLIS), to develop software that can exchange data, based on IFC 2.0 that support specific industry processes. One of the supported processes is design-to-building energy performance analysis. It is now possible to import building geometry from CAD tools via middleware into EnergyPlus 1.0. The middleware that serves as the link is BSPro Com-Server, developed by Olof Granlund OY in Finland [Karola and Lahtela 2000], and its EnergyPlus Client, developed at the Lawrence Berkeley National Laboratory. Both are bundled with EnergyPlus.

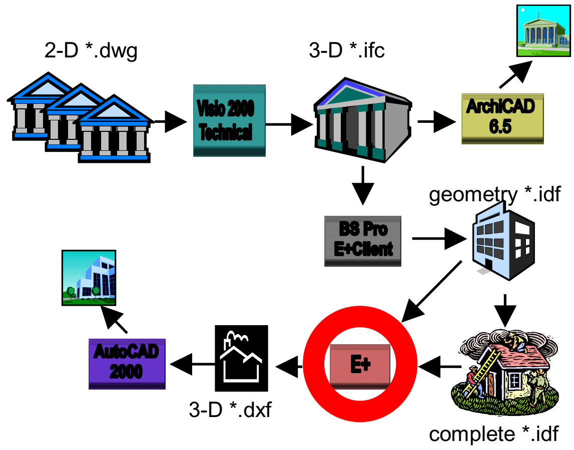

Figure 5 – Flow chart of the BLIS process of importing building geometry from CAD into EnergyPlus

The process is illustrated in Figure 5. If the building is only documented in 2-D, CAD files are imported into a CAD tool such as Visio 2000 Technical, ArchiCAD 6.5 or Bricsnet Architecturals to interactively add the third dimension and save it in the *.ifc file format. If the building is originally defined in 3-D, the process begins with the saving of data in the *.ifc file format. BS Pro Server imports the *.ifc file and the EnergyPlus Client extracts from it all building geometry definitions needed for simulation with EnergyPlus. The Client also arranges the information according to the rules and syntax of EnergyPlus and saves it in EnergyPlus input data format, *.idf. That file contains only building geometry information, but EnergyPlus can import it and verify its correctness: It generates a *.dxf file that can be displayed in any CAD tool that can import that format. The content of the geometry file can then be spliced into a file that contains the rest of the data needed to execute an EnergyPlus simulation.

POSSIBLE SAVINGS

The definition of geometry for the DOE-2 simulation of the David L. Lawrence Convention Center (Figure 4) was difficult and took more than three man-weeks to complete. One has to wonder how much could have been saved had it been possible to at least partially automate the process.

To get at least some quantitative understanding of the possible savings, BLIS partners developed a test building: a three-story office building of modest architectural complexity. 2-D floor plans (drawn with AutoCAD 2000) were imported in Visio 2000 Technical to modify the geometry for energy performance simulation and to extrude the entire building vertically. In this case Visio software served as a substitute for a GUI. The resulting *.ifc file was visually tested with ArchiCAD 6.5 (Figure 6) and then imported into BSPro. The BSPro Client for EnergyPlus then generated an input file for EnergyPlus that contained building geometry necessary for the simulation. An EnergyPlus simulation run of the input file with the created geometry and the resulting *.dxf file confirmed that the building geometry was imported correctly.

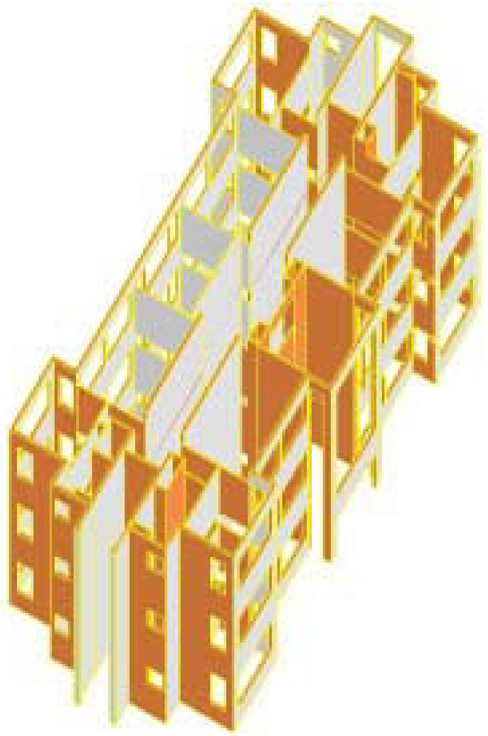

Figure 6 – BLIS office building; floor and roof slabs are removed to allow inspection of interior surfaces

The entire effort took less than three hours. In the opinion of the author, it would have taken 10-12 hours to define the geometry and import it manually. This indicates a savings ratio of approximately 1:4 for a building of modest size and complexity. Larger and more complex buildings should yield a higher ratio.

A number of architectural and engineering organizations have expressed interest in using these tools on real-life projects. BLIS partners will support such efforts. These projects will yield further understanding of the possible savings from acquisition of geometry using BLIS interoperable tools.

CONCLUSIONS

It is clear that the ideal of “seamless” acquisition of building geometry for building energy performance simulation is not at hand. The difference between the “simulation view” of the building and the view shared by architects and engineers is too varied and too significant. The concept of “pressing the button” to generate and import geometry is not realistic: Human intervention is unavoidable if the generated building geometry is to properly define the building to the simulation.

Still, tools exist today that can automate *parts* of the process that defines the definition and import of building geometry. These tools can expedite the process, avoid most errors and make the overall simulation effort more productive. Ultimately, this may result in a much more frequent use of building energy performance simulation.

REFERENCES

Bazjanac, V., and D.B. Crawley. 1999. “Industry Foundation Classes and Interoperable Commercial Software in Support of Design of Energy-Efficient Buildings,” *Proceedings of Building Simulation ’99*, Kyoto.

Haas, W. 2001. Personal communications.

Industry Alliance for Interoperability. 1999. *Introduction to the IAI and Its IFCs*, IAI.

Karola, A. and H. Lahtela. 2000. “BSPro Com-Server – Interoperability Between Software Tools Using Industry Foundation Classes,” *Building Energy Simulation User News*, Vol.21, No.5.

Naylor, Thomas H., J. L. Balintfy, D. S. Burdick and K. Chu. 1966. *Computer Simulation Techniques*, John Wiley & Sons, New York.